Information Fusion 92 (2023) 420–434

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

Enhanced Frequency Fusion Network with Dynamic Hash Attention for image denoising

Bo Jiang [a,1], Jinxing Li [a,1], Huafeng Li [d,2], Ruxian Li [e,3], David Zhang [b,4], Guangming Lu [a,c,*]

[a] Department of Computer Science and Technology, Harbin Institute of Technology Shenzhen, China
[b] School of Data Science, Chinese University of Hong Kong Shenzhen, China
[c] Guangdong Provincial Key Laboratory of Novel Security Intelligence Technologies, China
[d] Kunming University of Science and Technology, China
[e] Linklogis, Shenzhen, China

ARTICLE INFO

Keywords:
Image denoising
Transformer
Enhanced frequency fusion
Dynamic hash attention
Decomposition frequency

ABSTRACT

Recently, Transformer-based image denoising methods have achieved great progress in the image denoising task. However, these methods also lead to two problems: (a) Since noise can destroy texture or details in the image, the resulting tokens with low weight values may have a negative impact on the reconstructed denoised image (*i.e.*, there are artifacts in the reconstructed image, *etc*); (b) Frequencies in different domains are ignored, leading to the missing of textural details in the reconstructed image. To this end, we propose an **E**nhanced **F**requency **F**usion **Net**work (EFF-Net) with dynamic hash attention for image denoising, called EFF-Net. Specifically, to alleviate the impact of problem (a), we present the **D**ynamic **H**ash **A**ttention (DHA) module, which aims to effectively mitigate the negative impact of tokens with low weight values on image denoising performance; Furthermore, we start from the frequency perspective and design the **E**nhanced **F**requency **F**usion (EFF) module with **D**ecomposition **F**requency (DF) as the core component, which aims to achieve the separation and fusion of noisy image content in the frequency domain, and appropriately reconstruct the image content of different frequency components at different locations. The DHA and EFF modules are integrated into plug-and-play **A**daptive **F**requency **E**nhancement (AFE) transformer blocks to selectively recover different frequencies based on long-range pixel dependency. The extensive experiments endorse the effective, and superior performance of our EFF-Net for image denoising.

1. Introduction

Image denoising is the computer vision task of removing visual quality problems due to noise during image acquisition or transmission. Therefore, image denoising is a critical step in many computer vision tasks [1–5]. However, as an ill-posed inverse problem, image denoising is also extremely challenging [6,7]. Recently, Transformer-based methods achieve state-of-the-art image denoising performance, even faster than typical **C**onvolutional **N**eural **N**etworks (CNNs)-based methods [8–11]. Furthermore, Transformer-based image denoising models [12,13] can also address the problem of modeling long-range pixel dependencies that CNNs are not good at.

Although these Transformer-based image denoising models improve the efficiency and generalization performance of image denoising tasks, they also lead to two main problems: ***(a) Since noise can destroy the texture or details in the image, the tokens with low weight value generated from the noisy image through the attention mechanism in the traditional Transformer method may adversely affect the reconstructed denoised image (i.e., the reconstructed image contains artifacts, etc)***. *In the Transformer method of image denoising, the essence of the attention mechanism is to focus on the tokens that contribute greatly to the reconstructed image, while weakening the redundant or useless tokens of the reconstructed image, and the corresponding attention weight value of these tokens is low. However, tokens with low weight values may*

* Correspondence to: Department of Computer Science and Technology, Harbin Institute of Technology at Shenzhen, Shenzhen 518057, China.
E-mail addresses: jiangbo_PhD@outlook.com (B. Jiang), lijinxing158@gmail.com (J. Li), hfchina99@163.com (H. Li), liruxian@linklogis.com (R. Li), davidzhang@cuhk.edu.cn (D. Zhang), luguangm@hit.edu.cn (G. Lu).
[1] Bo Jiang and Jinxing Li are with the Department of Computer Science and Technology, Harbin Institute of Technology at Shenzhen, Shenzhen 518057, China.
[2] Huafeng Li is with Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, Yunnan 650500, PR China.
[3] Ruxian Li is with Linklogis, Shenzhen, 518071, Guangdong, PR China.
[4] David Zhang is with the School of Science and Engineering, The Chinese University of Hong Kong at Shenzhen, Shenzhen 518172, China.

https://doi.org/10.1016/j.inffus.2022.12.015
Received 24 October 2022; Received in revised form 27 November 2022; Accepted 16 December 2022
Available online 20 December 2022

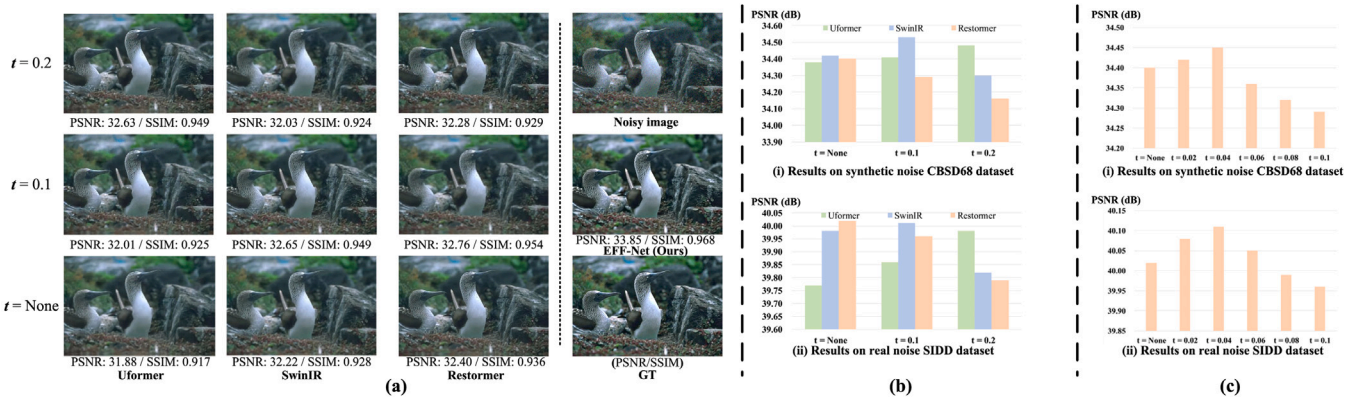

Fig. 1. Effect of tokens with low weight values on image denoising performance. (a) Visual effects on a single image. (b) Image denoising effects on synthetic noise datasets and real noise datasets.

still adversely affect the reconstructed denoised image. ***(b) Meanwhile, by omitting the use of different frequency domain layered features to enhance and reconstruct the texture or details of an image, the traditional Transformer-based denoising method may also cause an excessive smoothing effect.***

For the issue (a), we perform an experimental analysis on tokens with low weight values to explore its effect on reconstructing denoised images. Specifically, we use UFormer [12], SwinIR [13], and Restormer [14] as models for toy cases. In addition, three thresholds are set separately to set the low-weight flags to zero. The three thresholds are $t = None, t = 0.1$, and $t = 0.2$, where $t = None$ denotes the original model. As shown in Fig. 1(a), on Uformer, when $t = 0.2$, the objective indicators and visualization effects are the best. For SwinIR and Restormer, satisfactory denoising results are obtained when $t = 0.1$. To be convincing, we perform the above toy case experiments on the synthetic noise CBSD68 dataset and the real noise SIDD dataset, respectively. As shown in Fig. 1(b), we observe that using a threshold to set the value of low-weighted tokens in the network to zero results in better objective quantification results compared to $t = none$. Therefore, tokens with low weight values may negatively affect the reconstructed image. It is worth noting that the image denoising performance of Restormer seems to decay as the threshold t increases. The reason for this phenomenon is that the optimal threshold of Restormer is in the threshold interval range of $t \in (0, 0.1)$. To verify this conjecture, we re-divide the interval of threshold t for Restormer and further explore the impact of low-weight tokens in Restormer on image denoising performance. As shown in Fig. 1(c), when the threshold value $t = 0.04$, the image denoising performance of Restormer on the CBSD68 and SIDD test datasets is the best. This also illustrates that it is very important to choose an appropriate threshold t to zero the tokens with low weights to improve the image denoising performance.

For the issue (b), we first analyze and compare synthetic and real noise-clean image pairs and the difference between noise-clean image pairs from the perspective of frequency distribution. Specifically, we transform all noise-clean image pairs in the dataset (including CBSD68 and Set12) from the spatial domain visualization to the frequency domain visualization by the Fourier transform tool [15]. As shown in Fig. 2, each frequency curve represents the average frequency curve of each group of image data sets. Meanwhile, the upper and lower boundaries of the background region accompanying the frequency curve represent the maximum frequency curve and the minimum frequency curve of the corresponding image data set, respectively. As shown in Fig. 2(a), on the synthetic noise image dataset, we observe that the degradation from clean image to synthetic noise image exists in all frequency components, and the frequency components with the largest degradation difference are concentrated in the low-frequency components, intermediate-frequency components, and high-frequency components in order. In contrast, in Fig. 2(b), on the real noise image dataset, although the degradation of the real noise image also exists in all frequency components, the frequency components with the largest degradation difference are different. Particularly, the largest difference is concentrated in the high-frequency component, the intermediate-frequency components and low-frequency components in turn. Unlike the synthetic noise-clean image pair, which has a single and uniform synthetic degradation, the real noise-clean image pair is captured using a Digital Single-Lens Reflex (DSLR) camera, which usually contains various/complex non-Uniform real-world noise degradation. This is one of the reasons why those classical deep learning-based image denoising methods (DnCNN [6], FFDNet [7], *etc.*) cannot handle the real image noise problem well.

Based on the above analysis, we propose an **E**nhanced **F**requency **F**usion **Net**work (EFF-Net) with dynamic hash attention for image denoising. Specifically, first, to alleviate the impact of problem (a) on image denoising performance, we propose a **D**ynamic **H**ash **A**ttention (DHA) module. The DHA module performs dynamic zeroing of the weights of tokens through the Hash layer, which can effectively alleviate the negative impact of tokens with low weight values on the image denoising performance. On the other hand, to solve problem (b) from the frequency perspective, we design the **E**nhanced **F**requency **F**usion (EFF) module with **D**ecomposition **F**requency (DF) as the core component, which can adaptively capture the different frequency component information to achieve the separation and fusion of noisy image content in the frequency domain. Then, to achieve the dynamic reconstruction of image denoising, we further integrate the DHA and EFF modules into a plug-and-play **A**daptive **F**requency **E**nhancement (AFE) transformer block to selectively recover different frequencies by incorporating long-range pixel dependencies. Finally, building the proposed EFF-Net with the AFE block can effectively improve the generalization ability of the network and the quality of recovered images.

The contributions of this paper can be summarized as follows:

- We analyze the essential difference between synthetic noise image degradation and real noise image degradation from the perspective of low-weight tokens and different frequency components to answer the impact on image denoising tasks. Based on this analysis, an **E**nhanced **F**requency **F**usion **Net**work (EFF-Net) for image denoising is proposed.
- To achieve the dynamic reconstruction of image denoising, a plug-and-play **A**daptive **F**requency **E**nhancement (AFE) transformer block is proposed, which contains two technical innovations: (1) **D**ynamic **H**ash **A**ttention (DHA) module, which aims to effectively mitigate the negative impact of tokens with low weight values on image denoising performance; (2) the **E**nhanced **F**requency **F**usion (EFF) module with **D**ecomposition **F**requency (DF) as the core component, which aims to achieve the separation and fusion of noisy image content in the frequency domain,

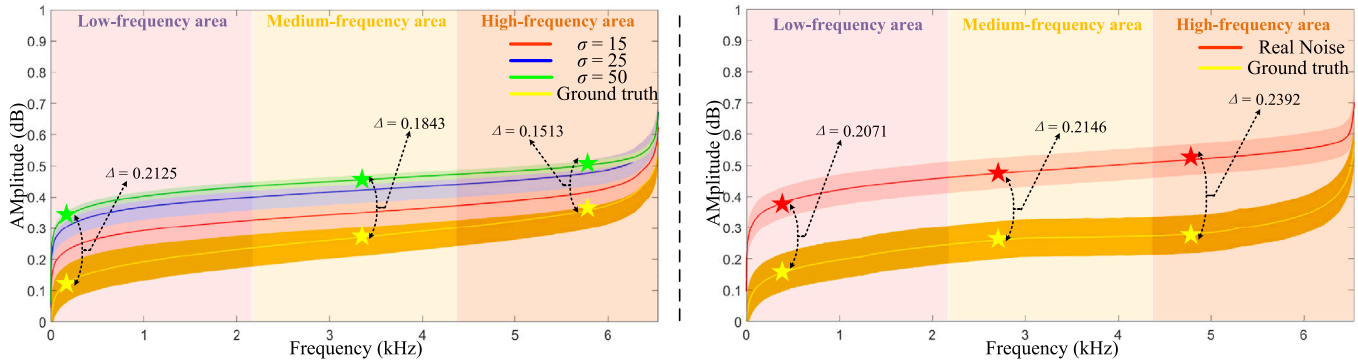

Fig. 2. Difference analysis in the frequency domain between the conventional synthetic noisy images and real noise images:(a) the statistics frequency distribution comparison in synthetic noisy image dataset with different noise levels. (b) the frequency components distribution comparison in real noise image dataset.

and appropriately reconstruct the image content of different frequency components at different locations.

- Extensive experiments on multiple image denoising benchmarks validate the effectiveness and superiority of our EFF-Net. Sufficiently intuitive means of visualizing and analyzing the functionality of the proposed DHA and EFF modules are also provided.

The paper is organized as follows: Section 2 briefly reviews related works on image denoising methods and Vision Transformer. Sections III presents the proposed Enhanced Frequency Fusion Network with Dynamic Hash Attention. Section 4 demonstrates the experiment results, and the paper is finally concluded in Section 5.

2. Related work

In the past few decades, the problem of image denoising has been extensively studied and remarkable progress has been made. We summarize and analyze various image denoising methods.

2.1. Image denoising

Among the traditional non-learning image denoising methods, most of them reconstruct the denoised image by modeling the noise distribution [16–19], so that the prior information can be manually designed. However, traditional non-learning methods for image denoising are limited due to the inflexibility of these hand-designed priors [20,21]. For instance, the internal and external non-local self-similarity (SNSS) priors are used for image denoising [22], the rank residual constraint (RRC) method was proposed for the image denoising [23]. Fortunately, with the advent of CNNs, the performance of deep CNN-based image denoising methods has greatly been improved in recent years. In recent years, a large number of excellent convolutional neural network-based image denoising methods have been proposed. DnCNN [6] introduces residual learning to reconstruct noise maps to obtain denoised images. FFDNet [7] adopts artificially set noise level to remove non-uniform noise in images. CBDNet [24] adopts a noise estimation sub-network to predict noise maps as prior knowledge to remove real noise in real scene images. RIDNet [25] enhances the flow of high-frequency information through an attention mechanism, and uses a residual structure to reduce the transfer of low-frequency information, thereby reconstructing detailed information in denoised images. MIRNet [26] proposes a multi-scale residual block to extract contextual features in low-resolution space to preserve details in denoised images. RDN [27] combines densely connected blocks and global residual learning to fuse local and global features for image denoising. Although these models try to improve the performance of image denoising, the receptive field of convolution operators has limitations for modeling long-range pixel dependencies, so the above methods may not easily overcome this limitation.

2.2. Vision transformer

Transformers started out in natural language pre-training tasks [28] and achieved amazing results on downstream tasks [29–32]. At present, Transformer has been successfully applied in the field of computer vision such as image recognition [33,34], segmentation [35,36], object detection [37,38]. Vision Transformers (ViT) [33,34] reshape an image into a series of flat 2D patch embeddings to learn the inter-relationships between them. This provides a strong basis for such models to learn long-range pixel dependencies between image patch sequences. Due to these properties, Transformer-based models have also been studied for image denoising tasks. IPT [39] is a neural network model built on the standard Transformer, which can be used for image denoising tasks. However, IPT is a pre-trained model, *i.e.*, the model first needs to be trained on a large-scale dataset, which means that the image denoising performance is hindered by the pre-training performance. Uformer [12] is a U-shaped network based on standard Transformer, which achieves good results in image denoising by modeling long-range pixel dependencies. SwinIR [13] proposes a model that can be used for image denoising, consisting of shallow feature extraction, deep feature extraction, and high-quality image reconstruction blocks. Restormer [14] can effectively remove noise from images by designing a multi-head attention and feed-forward network. SCUNet [40] combines the local modeling ability of the residual convolutional layer with the non-local modeling ability of the Swin-Transformer block [41], and then inserts it as the main building block into the U-shaped network architecture to achieve the purpose of noise removal. However, these methods also suffer from the following two main problems: first, since noise can destroy the texture or details in the image, the resulting labels with low weight values may negatively affect the reconstructed denoised image; second, previous the Transformer-based image denoising model ignores the use of layered features in different frequency domains to enhance and reconstruct the texture or details of the image, which may also lead to over-smoothing of the denoised image.

Different from previous Transformer-based image denoising methods, the proposed Enhanced Frequency Fusion Network (EFF-Net) with dynamic hash attention for image denoising has stronger robustness and generalization. Specifically, to alleviate the impact of the first problem, we propose a Dynamic Hash Attention (DHA) module, which aims to effectively alleviate the negative impact of tokens with low weight values on image denoising performance; Due to the influence of the second problem, we design the Enhanced Frequency Fusion (EFF) module with Decomposition Frequency (DF) as the core component from the perspective of frequency, which aims to separate and fuse the noise image content in the frequency domain and appropriately reconstruct the image content of different frequency components at different positions. Then, the DHA and FE modules are further integrated into a plug-and-play Adaptive Frequency Enhancement (AFE) transformer block to achieve the dynamic reconstruction of image denoising by incorporating long-range pixel dependencies.

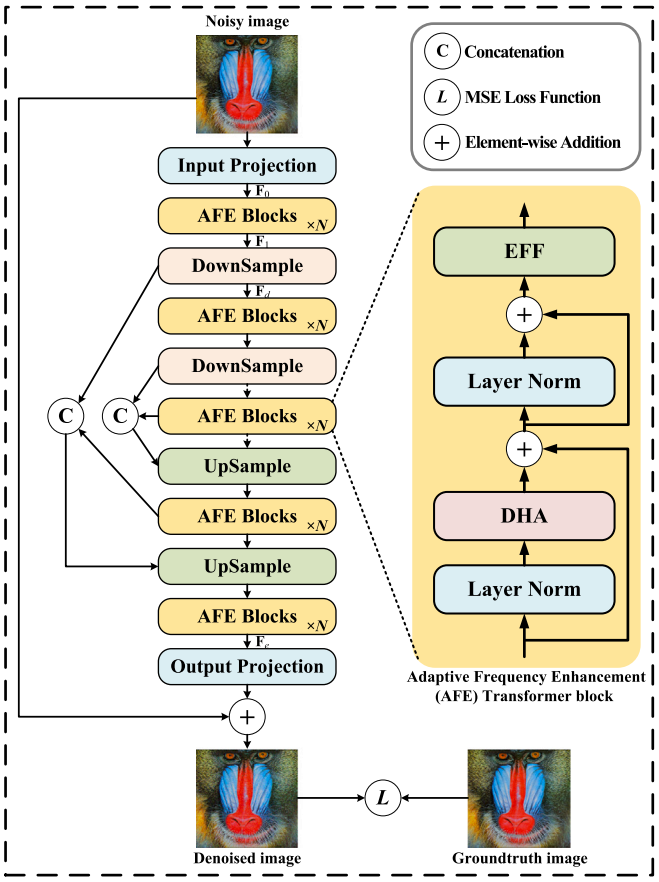

Fig. 3. (a) Overview of the EFF-Nets structure. (b) Structure of the Adaptive Frequency Enhancement (AFE) Transformer block.

3. Enhanced frequency fusion network

In this section, we first describe the overall structure of the Enhanced Frequency Fusion Network (EFF-Net) with dynamic hash attention for image denoising. Then, we introduce the details of the overall pipeline of EFF-Net, and the core components of Adaptive Frequency Enhancement (AFE) transformer block (*i.e.*, Dynamic Hashing Attention (DHA) and Enhanced Frequency Fusion (EFF)), respectively. Next, we discuss the effect of widely used frequency domain transformation methods on disassembling frequency and the difference between these transform methods and the proposed Disassemble Frequency (DF) (*i.e.*, the core component of EFF module). After that, we introduce the optimized loss function of EFF-Net.

3.1. Overall pipeline

As shown in Fig. 3, the overall structure of the proposed EFF-Net is a symmetric encoder–decoder architecture. Specifically, given a noisy image $\mathbf{I}_n \in \mathbb{R}^{H\times W\times 3}$, EFF-Net first employs the Input Projection layer to obtain the token embedding feature $\mathbf{F}_0 \in \mathbb{R}^{S\times C}$, where H and W are the height and width of the noise image, respectively. C is the number of channels generated by the Input Projection layer, $S = N\times W$ represents the size of each token. The token embedding feature extracted by the Input Projection layer (*i.e.*, 3 × 3 convolutional layer and LeakyReLU) is shown in Eq. (1):

$$\mathbf{F}_0 = \varphi\left(f_c^{in}\left(\mathbf{I}_n \in \mathbb{R}^{H\times W\times 3}\right)\right) \xrightarrow{\mathcal{R}} \mathbb{R}^{S\times C}, \tag{1}$$

where f_c^{in} denotes convolutional layer, φ represents LeakyReLU activation layer, and $\xrightarrow{\mathcal{R}}$ is a reshape operation on a tensor. Then, the low-level features $\mathbf{F}_0$ are fed into the encoder, which consists of Adaptive Frequency Enhancement (AFE) Transformer blocks and Downsampling layers. The AFE block is mainly composed of Dynamic Hashing Attention (DHA) and Enhanced Frequency Fusion (EFF). The AFE Transformer block takes advantage of the DHA mechanism to dynamically make zero of the weights of tokens through the Hash layer to alleviate the negative impact of tokens with low weight values. Additionally, it also enhances frequency, which is achieved by adaptively capturing different degenerate frequency component characteristic information on the feature maps through the EFF module. In the Downsampling layer, we reshape the token embedding feature into a 2-D spatial feature map, and then use a convolutional layer with padding of one, a stride of two, and a convolution kernel size of four to downsample and double the number of output channels. Formally, given the feature map $\mathbf{F}_1 \in \mathbb{R}^{S\times C}$ output by the AFE block, the Downsampling layer produces the feature map $\mathbf{F}_d$ as:

$$\mathbf{F}_d = \varphi\left(f_{\downarrow}\left(\mathbf{F}_1 \in \mathbb{R}^{S\times C} \xrightarrow{\mathcal{R}} \mathbb{R}^{C\times H\times W}\right)\right), \tag{2}$$

where $f_{\downarrow}$ is the downsampling function with a scaling factor of two, $\mathbf{F}_d \in \mathbb{R}^{2C\times\frac{H}{2}\times\frac{W}{2}} \xrightarrow{\mathcal{R}} \mathbb{R}^{\frac{S}{4}\times 2C}$. Hence, the latent feature generated after passing through the all encoders is $\mathbf{F}_{en} \in \mathbb{R}^{\frac{S}{16}\times 4C}$. The low-resolution spatial feature maps obtained from the encoder are refined using a AFE block as the bottleneck block. Next, the feature maps obtained from the encoder and bottleneck block are fed to the decoder, which consists of AFE blocks and Upsampling layers. To assist reconstruct the denoised image process, the encoder features and decoder features are concatenated through a concatenation operation. The concatenation operation is followed by an Upsampling layer to improve the spatial resolution of the feature maps and reduce the number of channels by half. Finally, after the decoder, we reshape the flattened features $\mathbf{F}_e$ into 2-D feature maps using an Output Projection layer (*i.e.*, 3 × 3 convolutional layer) is shown in Eq. (3):

$$\mathbf{I}_d = f_c^{out}\left(\mathbf{F}_e \in \mathbb{R}^{S\times C} \xrightarrow{\mathcal{R}} \mathbb{R}^{H\times W\times 3}\right) + \mathbf{I}_n, \tag{3}$$

where f_c^{out} denotes convolutional layer, and $\mathbf{I}_d$ represents the reconstructed image after denoising.

3.2. Dynamic hashing attention (DHA)

To alleviate the negative impact of tokens with low weight values on the reconstructed denoised images, we propose **D**ynamic **H**ash **A**ttention (DHA), which aims to zero out low-weight tokens by a learnable threshold in the Hash layer, gradually reducing its negative impact on the reconstructed image. Since the learnable threshold is generated by perceiving different noisy sample content, zeroing low-weight tokens is a dynamic process.

Specifically, as shown in Fig. 4(a), given the input $\mathbf{F}_{in}^D \in \mathbb{R}^{S\times C}$, we use the Pooling Layer to perform feature aggregation on $\mathbf{F}_{in}^D$, and these aggregated features are fed into the convolutional layers and the activation layer to further refine the features, as shown in Eq. (4):

$$\mathbf{F}_{\mathbf{w}} = f_c^2\left(\phi\left(f_c^1\left(\mathrm{PL}\left(\mathbf{F}_{in}^D\right)\right)\right)\right) \in \mathbb{R}^{1\times C}, \tag{4}$$

where PL denotes adaptive average pooling layer, both f_c^1 and f_c^2 represent the convolutional layers with convolution kernel 1 × 1, and ϕ is GeLU activation layer. Then, the refined features are passed through the Hash Layer to form the weight distribution of tokens, *i.e.*, the tokens with low weights are dynamically set to zero. The equation is shown as Eq. (5):

$$W_{i,j} = \begin{cases} F_{i,j}, F_{i,j} \in \mathscr{H}\left(\mathbf{F}_{\mathbf{w}}\right) \geq t \\ 0, \quad F_{i,j} \in \mathscr{H}\left(\mathbf{F}_{\mathbf{w}}\right) < t \end{cases} \in \mathbf{W}_H, \tag{5}$$

where $\mathscr{H}$ represents Hash (Tanh activation) layer, which is designed to compress features. Hashing refers to a function that compresses an information interval of any length into a certain fixed-length information interval [42]. Anything that conforms to this idea is called a hash

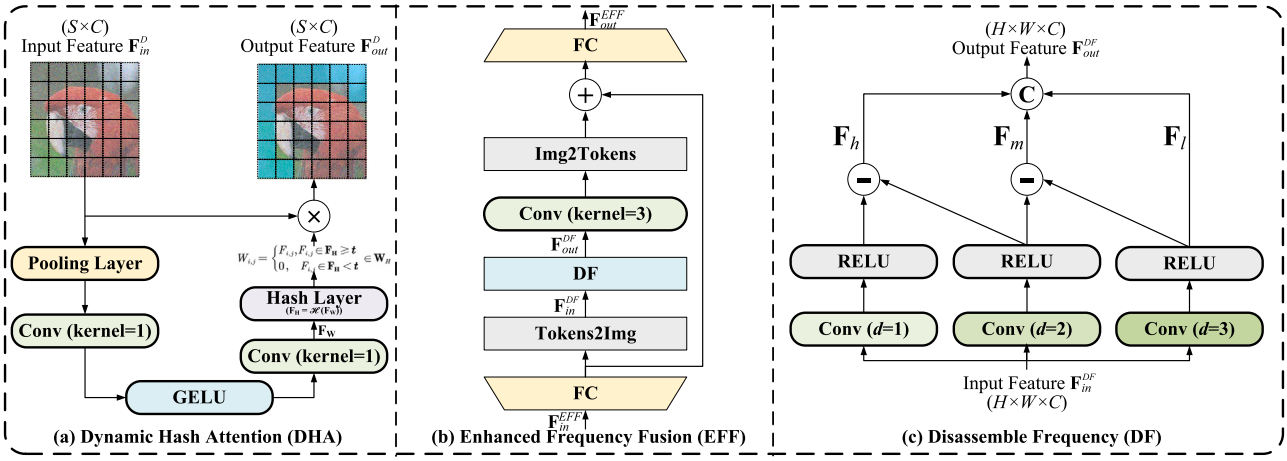

Fig. 4. (a) Structure of Dynamic Hash Attention (DHA). (b) Structure of Enhanced Frequency Fusion (EFF). (c) Structure of Disassemble Frequency (DF). ⊗ denotes the element-wise multiply operation, ⊖ is element-wise subtraction operation. d denotes dilated convolutions with different dilation rates.

algorithm [43]. Since the Tanh activation function (*i.e.*, $\mathscr{H}(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$, where z is input variable.) is an odd function, its function curve is strictly monotonically increasing through the origin and the first and third quadrants, its value range is limited between the two horizontal asymptotes $y = 1$ and $y = -1$. This conforms to the definition of Hash mapping [44], and the nonlinear Tanh activation function is convenient to implement. Thus, we use the nonlinear Tanh activation function to implement the Hash mapping. $F_{i,j}$ denotes the element in the ith row and jth column of $\mathbf{F}_{\mathbf{w}}$. $W_{i,j}$ represents the element in the ith row and jth column of the produced weight distribution $\mathbf{W}_H \in \mathbb{R}^{1\times C}$. The $\boldsymbol{t} \in \mathbb{R}^{1\times C}$ is a learnable threshold, and its the calculation process is shown in Eq. (6):

$$\boldsymbol{t} = f_{Linear}\left(\mathrm{PL}\left(\mathbf{F}_{\mathbf{W}}\right)\right), \tag{6}$$

where f_{Linear} represents the linear function layer. Since the weight of the linear function layer is learnable, the threshold $\boldsymbol{t}$ is a tensor value obtained by learning, which is called the learnable threshold $\boldsymbol{t}$. Inspired by the success of LatticeNet [45], we utilize the statistics of the feature maps to compute the learnable threshold $\boldsymbol{t}$: the mean of the feature maps, ***which is a widely used and enough statistical property to describe the feature maps.*** In other words, the threshold $\boldsymbol{t}$ is obtained by mean learning of the feature maps $\mathbf{F}_{\mathbf{W}}$, where $\mathrm{PL}\left(\mathbf{F}_{\mathbf{W}}\right)$ represents the mean of the feature maps $\mathbf{F}_{\mathbf{W}}$.

The final output features are obtained by the above zeroed weight $\mathbf{W}_H$ rescaling transformation:

$$\mathbf{F}_{out}^{D} = \mathbf{F}_{in}^{D} \cdot \mathbf{W}_H, \tag{7}$$

where $\mathbf{F}_{out}^{D} \in \mathbb{R}^{S\times C}$ is the DHA output feature maps.

The proposed EFF-Net is composed of an Input Projection layer, an Output Projection layer, Downsampling layers, Upsampling layers, and AFE Blocks. There are four groups of AFE Block and the Downsampling layer that constitute the encoder, while four groups of AFE Block and the Upsampling layer make up the decoder. For Downsampling and Upsampling layers, the downsampling and upsampling scale factors are both set to 2.

3.3. Enhanced frequency fusion (EFF)

To adaptively capture the characteristic information of different degenerate frequency components, we propose the Enhanced Frequency Fusion (EFF) module. It aims to decompose noisy image features into low-/ intermediate-/high-frequency components, and enhance and fuse different frequencies with local and global information. As shown in Fig. 4(b), the EFF module is mainly composed of two fully connected layers, a Disassemble Frequency (DF) module, and a convolutional layer.

It is well known that in natural images, high-frequency information usually corresponds to the texture and details in the image, while low-frequency information corresponds to the global structure of the image. ***Therefore, convolutional layers with large receptive fields are used to capture coarse features, i.e. low-frequency information. Then, we remove such low-frequency components from the original feature (after atrous convolution) to obtain the remainder relatively high-frequency features.*** Based on this, inspired by [46], as shown in Fig. 4(c), we propose the Disassemble Frequency (DF) module, which aims to separate low-/ intermediate-/high-frequency components from the input feature maps. Specifically, we first use a dilated convolutional layer [47] with a dilation rate $d = 3$ to extract features with a large receptive field from the input feature maps to approximate the structural information, i.e. as low-frequency components. Then, we use a convolutional layer with a dilation rate of $d = 2$ to extract finer features from the input feature maps, and remove low-frequency components from the obtained features to obtain higher-frequency features. Finally, the separated features of different frequency components are spliced in the channel dimension.

Mathematically, given the EFF module input feature maps $\mathbf{F}_{in}^{EFF} \in \mathbb{R}^{S\times C}$, we use the fully-connected layer to extract global feature from $\mathbf{F}_{in}^{EFF}$, as shown in Eq. (8):

$$\mathbf{F}_{in}^{DF} = \phi\left(f_l^1\left(\mathbf{F}_{in}^{EFF}\right) \xrightarrow{\mathcal{R}} \mathbb{R}^{C\times H\times W}\right), \tag{8}$$

where $\mathbf{F}_{in}^{DF} \in \mathbb{R}^{C\times H\times W}$ denotes input feature map of the DF module, f_l^1 represent the fully-connected layer. Then, these feature maps are fed into the DF module to separate low-/intermediate-/high-frequency components, as shown in Eq. (9):

$$\begin{cases} \mathbf{F}_l = \sigma\left(f_{dc}^{d=3}\left(\mathbf{F}_{in}^{DF}\right)\right) \\ \mathbf{F}_m = \sigma\left(f_{dc}^{d=2}\left(\mathbf{F}_{in}^{DF}\right)\right) - \mathbf{F}_l \\ \mathbf{F}_h = \sigma\left(f_{dc}^{d=1}\left(\mathbf{F}_{in}^{DF}\right)\right) - \mathbf{F}_m \end{cases}, \tag{9}$$

where $\mathbf{F}_l$, $\mathbf{F}_m$, and $\mathbf{F}_h (\in \mathbb{R}^{\frac{C}{3}\times H\times W})$ denote low-/intermediate-/high-frequency feature components, respectively. $f_{dc}^{d=3}$, $f_{dc}^{d=2}$, and $f_{dc}^{d=1}$ represent dilated convolutional layers with dilation rates of $d = 3$, $d = 2$, and $d = 1$, respectively. σ is ReLU activation layer. To efficiently enhance each frequency feature, we adopt a depth-wise convolutional layer and a fully-connected layer to enhance and retrieval the low-/intermediate-/high-frequency feature components. The equation is shown as Eq. (10):

$$\begin{aligned} \mathbf{F}_{\text{out}}^{EFF} &= f_l^2\left(\varphi\left(f_{dp}\left(\mathbf{F}_{\text{out}}^{DF}\right) \xrightarrow{\mathcal{R}} \mathbb{R}^{S\times C}\right) + f_l^1\left(\mathbf{F}_{in}^{EFF}\right)\right) \\ &= f_l^2\left(\varphi\left(f_{dp}\left(P_{\text{con}}\left(\mathbf{F}_l, \mathbf{F}_m, \mathbf{F}_h\right)\right) \xrightarrow{\mathcal{R}} \mathbb{R}^{S\times C}\right) + f_l^1\left(\mathbf{F}_{in}^{EFF}\right)\right) \end{aligned}, \tag{10}$$

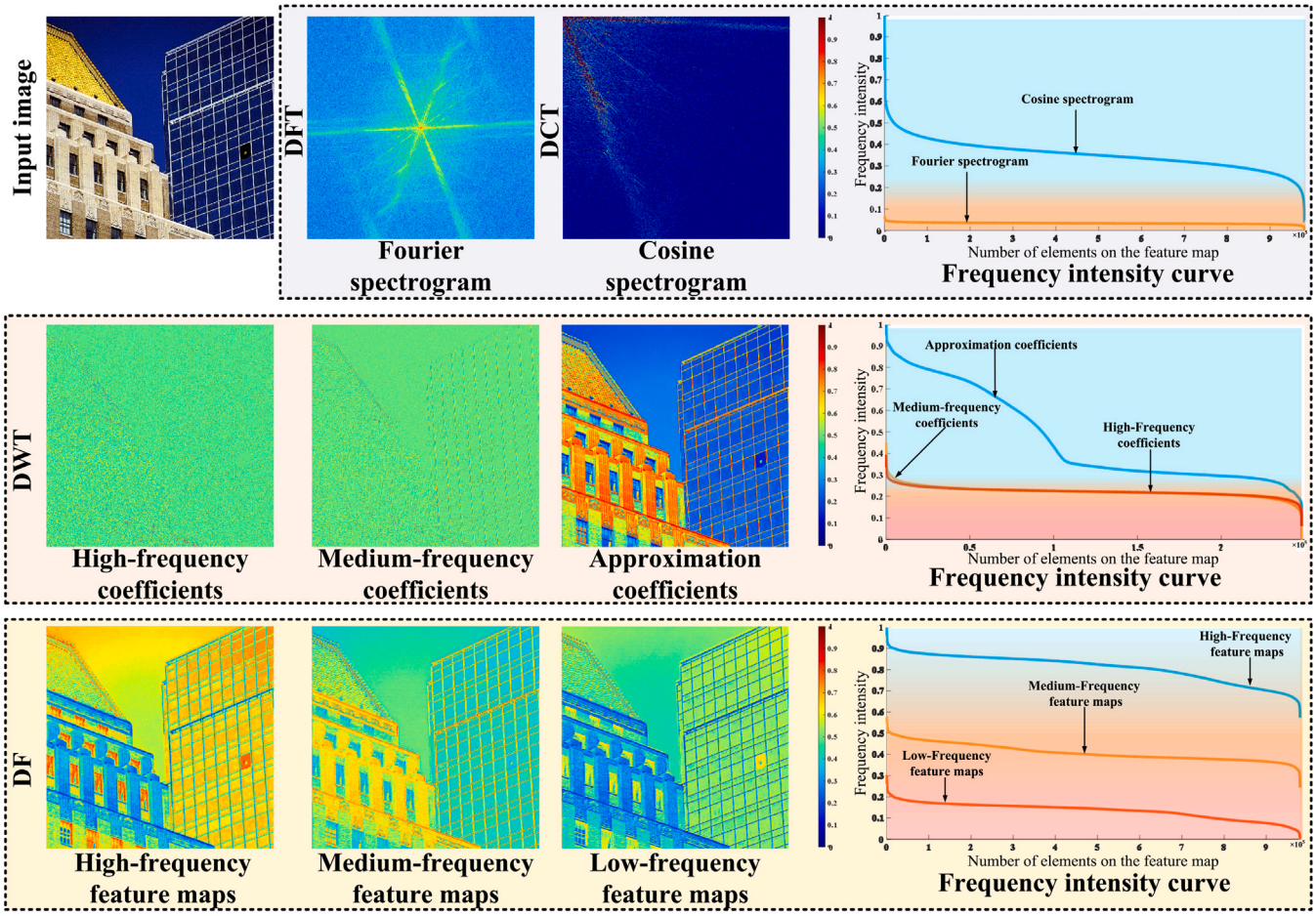

Fig. 5. Visualization of image transformation from the spatial domain to the frequency domain using DFT, DCT, DWT, and DF module and the corresponding frequency intensity curves. Number of elements on the feature map represents the corresponding frequency intensity of the nth element is $k \in (0, 1)$).

where $\mathbf{F}_{\text{out}}^{EFF} \in \mathbb{R}^{C\times H\times W}$ and $\mathbf{F}_{\text{out}}^{DF} \in \mathbb{R}^{C\times H\times W}$ denote output feature map of the EFF and DF module, respectively. P_{con} represents the concatenation operation. f_l^2 and f_{dp} are the fully-connected layer and a depth-wise convolutional layer, respectively.

3.4. Disassemble frequency discussion

Currently, the commonly used methods for transforming an image from the spatial domain to the frequency domain are Discrete Fourier Transform (DFT) [48], Discrete Cosine Transform (DCT) [49] and Discrete Wavelet Transform (DWT) [50]. As shown in Fig. 5, both DFT and DCT characterize the frequency domain characteristics of the entire spatial domain of the image in the frequency domain, *i.e.*, the spectrogram of the full frequency band, it is not conducive to the frequency at which low- / medium- / high-components are decomposed. DWT can not only characterize the frequency domain features of the entire spatial domain of the image in the frequency domain, but also can characterize the frequency domain features of the local spatial domain. A series of wavelet coefficients, including high-frequency coefficients, intermediate-frequency coefficients, and approximate coefficients (*i.e.*, low-frequency coefficients), can be obtained directly from the input image through a first-order DWT. However, in the frequency intensity curve, it is found that the approximate coefficients contain a lot of high-frequency information. At the same time, there is not enough discrimination between the high-frequency coefficients and the intermediate-frequency coefficients. Therefore, the above-mentioned deterministic mathematical calculation methods (DFT, DCT, and DWT) cannot flexibly perform frequency decomposition operations according to input samples.

To flexibly address the frequency decomposition problem according to the input samples, we propose a DF module implemented only using dilated convolutions with different dilation rates. As shown in Fig. 5, in the DF module, the high-/intermediate-/low-frequency feature maps are separated from the input image using dilated convolutions with dilation rates $d = 1$, $d = 2$, and $d = 3$, respectively. It is worth noting that the frequency feature map here refers to the feature map in the spatial domain. To more intuitively observe the degree of discrimination of frequency components on the frequency intensity curve, we employ the DFT to convert the high-/intermediate-/low-frequency feature maps into frequency domain space for frequency band comparison. From the frequency intensity curve in Fig. 5, it is found that the high-/intermediate-/low-frequency feature maps decomposed by the DF module have a greatly significant degree of discrimination of frequency components in the frequency domain space. Therefore, compared with DFT, DCT, and DWT, the proposed DF module can provide effective feature maps of each frequency component for adaptive frequency enhancement, thereby improving the performance of EFF-Net in removing noise in noisy images.

3.5. Loss function

The proposed EFF-Net uses clean-noisy paired images for a supervised training strategy. At the same time, in order to effectively prevent the model from overfitting, we use Charbonnier loss [51] as the loss function of our EFF-Net. The specific calculation equation is as follows:

$$\mathcal{L} = \frac{1}{N}\sum \sqrt{\|f_{EFF}\left(\mathbf{I}_n\right) - \mathbf{I}_g\|^2 + \zeta^2}. \tag{11}$$

Table 1
Average PSNR of the denoised grayscale images from Set12, BSD68 and Urban100 datasets. The values of PSNR are positively correlated with visual quality.

Datasets	σ	DnCNN	IRCNN	FFDNet	MWCNN	NLRN	RNAN	FOCNet	DAGL	DRUNet	SwinIR	SCUNet	Restormer	EFF-Net-T	EFF-Net-S	EFF-Net-B
Set12	15	32.86	32.77	32.75	33.15	33.16	–	33.07	33.28	33.25	33.36	33.43	33.42	33.36	33.41	33.49
	25	30.44	30.38	30.43	30.79	30.80	–	30.73	30.93	30.40	31.01	31.09	31.08	30.81	31.10	31.14
	50	27.18	27.14	27.32	27.74	21.64	27.70	27.68	27.81	27.90	27.91	28.04	28.00	27.92	28.06	28.13
BSD68	15	31.73	31.63	31.63	31.88	31.88	–	31.83	31.93	31.91	31.97	31.99	31.96	31.92	32.01	32.08
	25	29.23	29.15	29.19	29.41	29.41	–	29.38	29.46	29.48	29.50	29.55	29.52	29.49	29.56	29.62
	50	26.23	26.19	26.29	26.53	26.47	26.48	26.50	26.51	26.59	26.58	26.67	26.62	26.61	26.69	26.77
Urban100	15	32.64	32.46	32.40	33.17	33.45	–	33.15	33.79	33.40	33.70	33.88	33.79	33.73	33.81	33.90
	25	29.95	29.80	29.90	30.66	30.94	–	30.64	31.39	31.11	31.30	31.58	31.46	31.45	31.57	31.65
	50	26.23	26.22	26.50	27.42	21.49	27.65	27.40	27.97	27.96	27.98	28.56	28.29	28.49	28.58	28.71

where N denotes the number of training samples, f_{EFF} stands for the function of the proposed EFF-Net. $\mathbf{I}_g$ represents the groundtruth corresponding to the input noise image $\mathbf{I}_n$, and ζ^2 is a constant that is empirically set to 1×10^{-6}.

4. Experiments

In this section, we first describe the implementation details of the proposed EFF-Net. Next, we set up the architectural scale of the EFF-Net. Then, we evaluate the performance of our EFF-Net on synthetic and real datasets and compare it with state-of-the-art methods. Finally, an ablation study is conducted on the EFF-Net to verify its effectiveness and superiority.

4.1. Implementation

There are 128 patches in each input training batch of the proposed EFF-Net. Each patch is the size of 256×256. We adopt the AdamW [52] optimizer whose β_1 and β_2 are set to 0.9 and 0.999 respectively. The initial learning rate is 1×10^{-4}. The network parameters are initialized using the Kaiming method in [53].

4.2. Architectural scales

To illustrate the effectiveness of the proposed EFF-Net architectural scale in image denoising, our experiments used three different scale parameters for EFF-Net. Therefore, we adjust different EFF-Net architectural scales by changing the number of feature channels C, and other settings remain unchanged. For instance, the depth N of the EFF block is set to 2. The three specific parameters of different scales are set as follows: EFF-Net-T (Tiny, $C = 18$), EFF-Net-S (Small, $C = 36$), and EFF-Net-B (Basic, $C = 54$).

4.3. Experiment comparisons

To evaluate the performance of the proposed EFF-Net for image denoising, our method is compared with the state-of-the-art image denoising methods in terms of quantitative and qualitative aspects. The proposed EFF-Net is evaluated on synthetic noisy image datasets (including grayscale and color image datasets) and real noisy image datasets for the comparative experiments. To provide a fair comparison, we present results of state-of-the-art image denoising methods, using publicly available implementations from the corresponding literature.

Synthetic grayscale noisy images. We evaluate the image denoising performance on grayscale image datasets (Set12 [6], BSD68 [54], and Urban100 [55]) with different levels of noise (σ = 15, 25, 50). Table 1 reports the results of denoising synthetic noisy images on three grayscale image test datasets. The proposed EFF-Net is compared with twelve state-of-the-art denoising methods, including DnCNN [6], IRCNN [56], FFDNet [57], MWCNN [58], NLRN [59], RNAN [60], FOCNet [61], DAGL [62], DRUNet [63], SwinIR [13], Restormer [14], EDT-B [64], and SCUNet [40]. For the three grayscale image datasets with different noise levels (σ = 15, 25, 50), the proposed EFF-Net achieves significantly better PSNR results than twelve state-of-the-art methods. Specifically, when different noise levels are evaluated, the lightweight EFF-Net-S gains similar PSNR to SCUNet. By comparison, PSNR gained by EFF-Net-B is higher than that obtained by comparison methods. Moreover, compared to SCUNet, EFF-Net-B averagely achieves 0.20 dB, 0.26 dB, and 0.24 dB improvement of PSNR on Set12, BSD68, and Urban100. This indicates the effectiveness of the proposed DHA and EFF modules in EFF-Net for denoising synthetic noisy grayscale images.

Additionally, more visual comparisons on different grayscale image datasets with σ = 15, 25, 50 are shown in Fig. 6. In comparison with other twelve state-of-the-art methods, our method can recover images from synthetic noisy images more satisfactorily without introducing over-smoothness or apparent artifacts. For example, the contours of leaves on tree branches, stripes on tiger bodies, and pattern textures on roofs reconstructed with EFF-Net are more clearly visible than other comparison methods. This indicates that the above methods have a bottleneck in texture and contour details (*i.e.*, high-frequency information) reconstruction. Inversely, the proposed EFF-Net uses the DHA module to reduce low-weight tokens to avoid introducing image artifacts, and can enhance the respective frequency band information through the EFF module to produce more satisfactory visualization results.

Synthetic color noisy images. Color image datasets (CBSD68 [54], Kodak24 [65], McMaster [66], and Urban100 [55]) with different noise levels (σ = 15, 25, 50) is used to evaluate the performance of our method for image denoising. Results of synthetic noisy image denoising on four color image test datasets are shown in Table 2. The methods compared include BM3D [18], DnCNN [6], IRCNN [56], FFDNet [57], BRDNet [67], DRUNet [63], SwinIR [13], Restormer [14], EDT-B [64], and SCUNet [40]. The proposed EFF-Net significantly outperforms ten state-of-the-art methods on the three color datasets with different noise levels (σ = 15, 25, 50). Specifically, the extremely lightweight EFF-Net-T achieves similar PSNR gains as SCUNet and SwinIR on the three color datasets. With an increase in channels, the lightweight EFF-Net-S outperforms the ten state-of-the-art methods in terms of PSNR. Furthermore, the proposed EFF-Net-B averagely achieves 0.16 dB, 0.25 dB, 0.12 dB, and 0.10 dB improvement of PSNR on CBSD68, Kodak24, McMaster, and Urban100 as compared to SCUNet. It is evident that using the proposed EFF-Net with different architectural scales can effectively remove the noise from the color noisy image.

Fig. 7 shows the visual comparisons of state-of-the-art methods on color noisy image dataset with σ = 25, 50. By comparison with these methods, the proposed EFF-Net reconstructs finer outlines and produces clearer visualization results. For instance, the angular lines of the windows on the roof and the silhouettes of pink lamps hanging from the sails reconstructed by EFF-Net are more clearly visible than other comparison methods. This shows that the proposed EFF-Net is effective in enhanced the image detail information when removing noise from color noise images through the DHA and EFF modules, resulting in more satisfactory visualization results.

Real-World Noisy Images. Real-world images usually contain a complex noise due to multiple sources of intricate noise. Therefore, it is crucial to evaluate the performance of the proposed EFF-Net on real

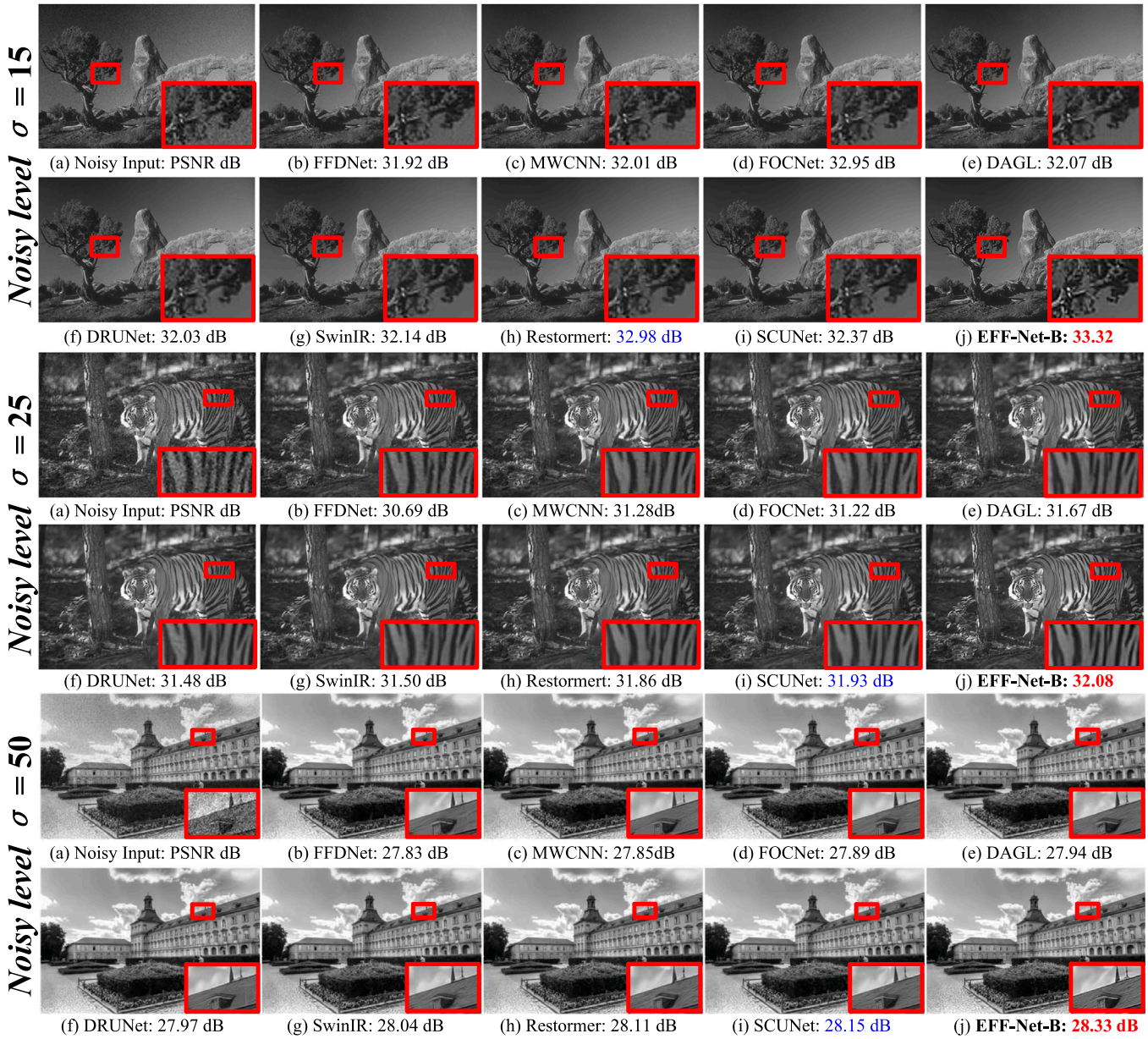

Fig. 6. Visual comparisons between EFF-Net and its competitors in the evaluation of grayscale noisy image denoising. The test images were selected from Set12, BSD68 and Urban100 with different noisy levels of σ=15, 25, 50.

Table 2
Average PSNR of the denoised color images from CBSD68, Kodak24, McMaster and Urban100 datasets. The values of PSNR is positively correlated with visual quality.

Dataset	σ	BM3D	DnCNN	IRCNN	FFDNet	BRDNet	DRUNet	SwinIR	Restormer	EDT-B	SCUNet	EFF-Net-T	EFF-Net-S	EFF-Net-B
CBSD68	15	33.52	33.90	33.86	33.87	34.10	34.30	34.42	34.40	34.39	34.40	34.39	34.46	34.72
	25	30.71	31.24	31.16	31.21	31.43	31.69	31.78	31.79	31.76	31.79	31.75	31.81	31.86
	50	27.38	27.95	27.86	27.96	28.16	28.51	28.56	28.60	28.56	28.61	28.55	28.63	28.71
Kodak24	15	34.28	34.60	34.69	34.63	34.88	35.31	35.34	35.47	35.37	35.34	35.30	35.48	35.54
	25	32.15	32.14	32.18	32.13	32.41	32.89	32.89	33.04	32.94	32.92	32.91	33.16	33.25
	50	28.46	28.95	28.93	28.98	29.22	29.86	29.79	30.01	29.87	29.87	29.93	30.08	30.10
McMaster	15	34.06	33.45	34.58	34.66	35.08	35.40	35.61	35.61	35.61	35.60	35.51	35.62	35.73
	25	31.66	31.52	32.18	32.35	32.75	33.14	33.20	33.34	33.34	33.34	33.33	33.40	33.48
	50	28.51	28.62	28.91	29.18	29.52	30.08	30.22	30.30	30.25	30.29	30.22	30.31	30.39
Urban100	15	33.93	32.98	33.78	33.83	34.42	34.81	35.13	35.13	35.22	35.18	35.11	35.19	35.27
	25	31.36	30.81	31.20	31.40	31.99	32.60	32.90	32.96	33.07	33.03	33.01	33.09	33.16
	50	27.93	27.59	27.70	28.05	28.56	29.61	29.82	30.02	30.16	30.14	30.05	30.18	30.21

Table 3
Average PSNR of the denoised real images from Nam, PolyU and SIDD datasets. The values of PSNR and SSIM are positively correlated with visual quality.

Dataset		DnCNN-B	FFDNet	TWSC	CBDNet	RIDNet	VDN	GCDN	PAN-Net	AINDNet	APD-Nets	MIRNet	HPDNet	Uformer	Restormer	EFF-Net-T	EFF-Net-S	EFF-Net-B
Nam	PSNR	36.08	37.85	38.37	38.51	38.72	39.16	38.96	40.18	39.21	40.36	39.88	40.26	–	–	39.98	40.12	40.27
	SSIM	0.903	0.938	0.952	0.957	0.960	0.965	0.962	0.978	0.966	0.989	0.973	0.979	–	–	0.974	0.979	0.981
PolyU	PSNR	35.74	37.19	37.63	37.85	38.07	38.43	38.21	39.91	38.78	–	39.25	39.89	–	–	39.87	40.01	40.13
	SSIM	0.878	0.939	0.954	0.956	0.957	0.960	0.958	0.971	0.963	–	0.971	0.970	–	–	0.968	0.970	0.978
SIDD	PSNR	38.56	38.60	35.89	38.68	38.71	39.29	38.93	39.33	39.45	39.75	39.71	39.72	39.77	40.02	39.79	39.98	40.15
	SSIM	0.910	0.909	0.838	0.909	0.913	0.911	0.910	0.912	0.915	0.959	0.959	0.958	0.959	0.960	0.957	0.959	0.964

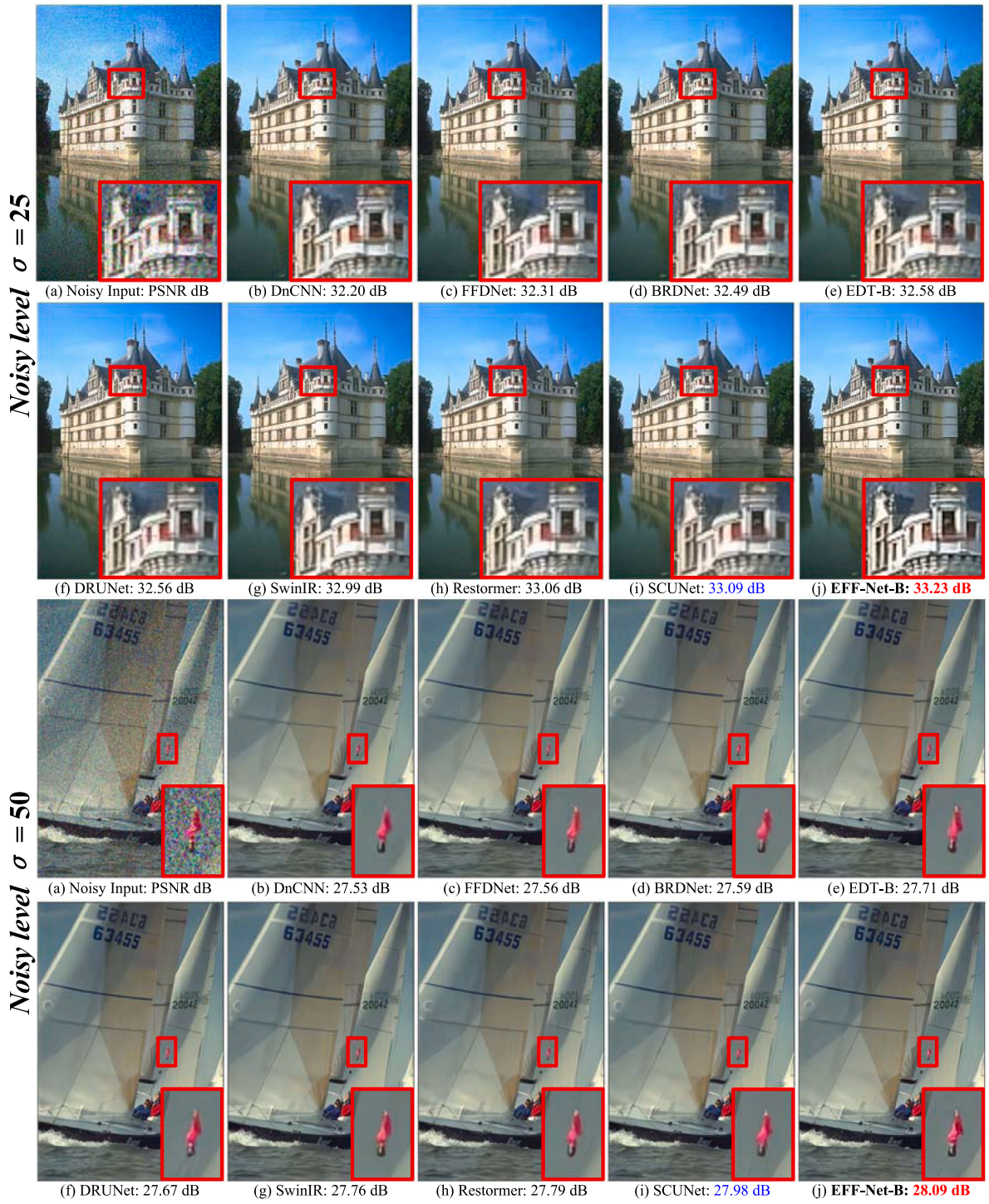

Fig. 7. Visual comparisons between EFF-Net and its competitors in the evaluation of color noisy image denoising. The test images were selected from CBSD68, Kodak24, McMaster, and Urban100 with different noisy levels of σ=25, 50.

noisy image datasets (here, SSID [68], PolyU [69], and Nam [70] are exploited). In this section, the proposed EFF-Net is compared to fourteen state-of-the-art denoising methods, including DnCNN-B [6], IRCNN [56], FFDNet [57], TWSC [71], CBDNet [24], RIDNet [25], VDN [72], PAN-Net [73], AINDNet [74], APD-Nets [20], Uformer [12], Restormer [14], MIRNet [26], and HPDNet [75]. Table 3 shows the results of real noisy image denoising performed on three real image datasets. On the SSID [68], PolyU [69], and Nam [70] real noisy image datasets, the proposed EFF-Net greatly increases the PSNR/SSIM results over the fourteen other state-of-the-art methods. The PSNR/SSIM gained by extremely lightweight EFF-Net-T are similar to these of Uformer and APD-Net on the SIDD dataset [68]. Additionally, it also performs competitively with MIRNet on PolyU datasets, and with PAN-Nets on Nam datasets. Furthermore, compared with MIRNet and HPDNet, EFF-Net-S outperforms them by an average PSNR of 0.24 dB on PolyU and 0.12 dB on Nam, respectively. Meanwhile, the proposed

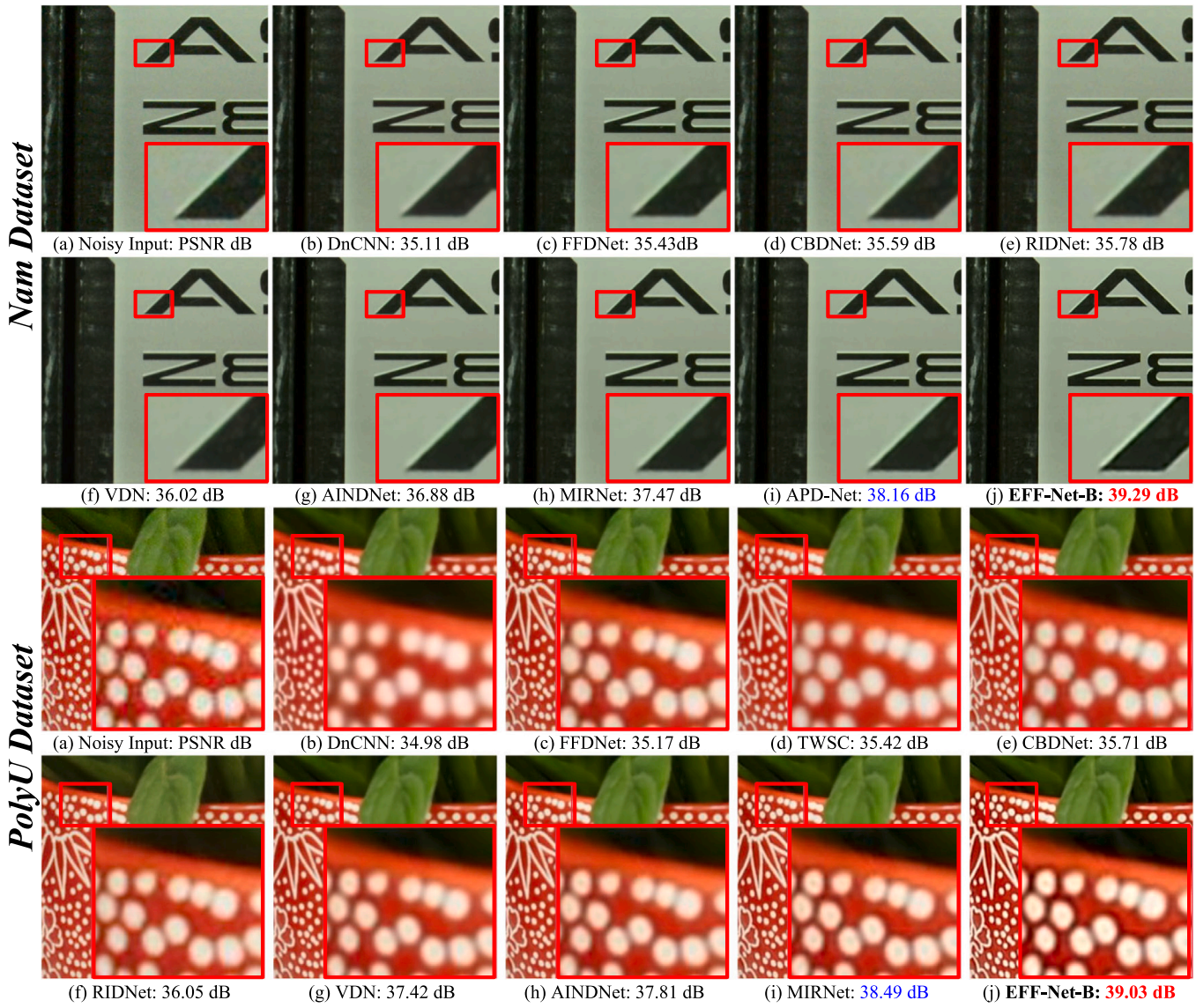

Fig. 8. Visual comparisons between EFF-Net and its competitors in the evaluation of real-noisy image denoising.

EFF-Net-B outperforms Restormer on SIDD by an average of 0.13 dB. This demonstrates that the proposed EFF-Net network structure is effective for denoising real noisy images. This further demonstrates the effectiveness of the proposed DHA and FE modules on real-world noisy image datasets.

To visually demonstrate the superiority of the proposed EFF-Net, Fig. 8 shows a visual comparison of state-of-the-art methods on PolyU [69] and Nam [70] datasets of real noisy images. As it turns out, our EFF-Net produces the most perceptually appealing visual quality among the competitive methods. Further, compared to other state-of-the-art methods, EFF-Net is better at removing unknown noise to achieve a visually pleasant appearance. As shown in Fig. 8, the outline of the book covers letter and the texture of the flower pot. Since the recent best five competing methods in Fig. 8 fail to handle unexpected and unseen noise, the recovered images suffer from blurring artifacts. This further demonstrates that the dynamic hashing attention mechanism and adaptively frequency enhancement can improve image denoising performance.

Efficiency of FAE-Net. To verify the efficiency of the proposed EFF-Net, we report the Runtime, FLOPs, and #Prams. of the compared state-of-the-art methods in Table 4. Meanwhile, for the fairness of the comparison, the results from all the compared methods in this table are obtained by inferring an image with the size of 256 × 256 on the same GPU device (*i.e.*, an Nvidia RTX Titan GPU). Our FAE-Net-T possesses the lowest FLOPs. Due to the self-attention mechanism, SwinIR, Restormer, and Uformer suffer from high FLOPs and long Runtimes. Furthermore, both DRUNet and SCUNet use a large number of parameters to improve image denoising performance, also leading to high FLOPs and long inference times. In contrast, our FAE-Net has the best balance between the FLOPs and inference time. Note that FAE-Net can effectively improve the computational efficiency by changing the architecture scale (*i.e.*, number of channels) while maintaining the image denoising performance well. Considering the denoising performance and efficiency, the proposed FAE-Net has extraordinary advantages over recent leading denoisers.

4.4. Ablation study

All the components in EFF-Net are quantitatively and qualitatively evaluated from two perspectives: (1) the DHA module; (2) the EFF module. Note that we adapt the EFF-Net-B (marked as EFF-Net) for ablation study. The synthetic color noisy image dataset with $\sigma = 15$ and real noisy image dataset (*i.e.*, Urban100 [55] and SIDD [68] validation dataset) are used for all ablation study experiments.

Table 4
Runtime, FLOPs, and #Prams. comparisons on image sizes of 256 × 256 on an Nvidia RTX Titan GPU with state-of-the-art methods. The results from all the compared methods in this table are obtained by inferring an image with the size of 256 × 256 on the same GPU device (*i.e.*, an Nvidia RTX Titan GPU).

Method	DRUNet	SwinIR	Uformer	SCUNet	Restormer	FAE-Net-T (Ours)	FAE-Net-S (Ours)	FAE-Net-B (Ours)
Runtime	0.092 s	0.132 s	0.254 s	0.098 s	0.153 s	**0.051 s**	0.073 s	0.080 s
FLOPs	143.63 G	503.95 G	84.76 G	79.84 G	140.9 G	**1.96 G**	7.18 G	15.68 G
#Prams.	32.64 M	7.77 M	39.56 M	17.95 M	26.13 M	**1.10 M**	4.34 M	9.70 M

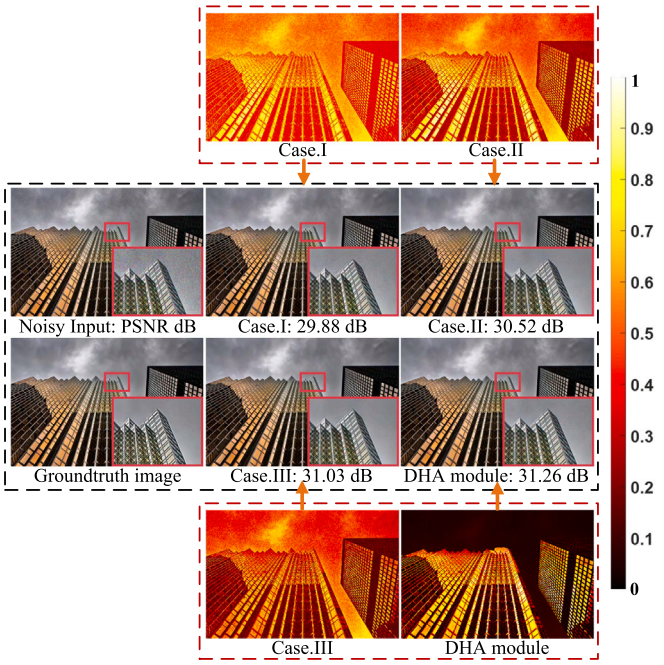

Fig. 9. Visual comparisons between EFF-Net and its competitors in the evaluation of image denoising.

Table 5
Performance effect of the overall structure of EFF-Net on image denoising. The PSNR (dB) is used as a reference metric for image denoising performance on the validation dataset.

DHA module		✓		✓
EFF module			✓	✓
Urban100 (PSNR:dB)	33.06	34.75	34.93	**35.27**
SIDD (PSNR:dB)	38.59	39.21	39.87	**40.15**

Table 6
Performance effect of the DHA module on image denoising. The PSNR (dB) is used as a reference metric for image denoising performance on the validation dataset.

Method	FLOPs	#Prams.	Urban100	SIDD
Case.I: Vanilla convolutional	51.33 G	21.30 M	34.53	39.36
Case.II: Channel attention	20.96 G	10.74 M	34.98	39.81
Case.III: Spatial attention	20.64 G	10.37 M	35.06	39.94
EFF-Net (Ours)	**20.37 G**	**10.28 M**	**35.27**	**40.15**

4.4.1. Ablation study of the overall structure of EFF-net

To study the gain of the overall structure of EFF-Net on image denoising, we conduct ablation experiments on the whole block of the DHA module and EFF module. As shown in Table 5, compared with the image denoising performance without DHA and EFF module, the image denoising performance of using both DHA and EFF module in EFF-Net is significantly improved. This shows that the DHA and EFF modules are the reasons why EFF-Net improves the performance gain of image denoising. For detailed ablation experiments of DHA and EFF modules, please refer to Sections 4.4.2 and 4.4.3, respectively.

4.4.2. Ablation study of the DHA module

To study the impact of dynamic hashing attention on image denoising, ***we set up the DHA module in EFF-Net and use vanilla convolutional layers, channel attention module*** [76] ***and spatial attention module*** [77] ***to replace the DHA module scheme (denoted as Case.I, Case.II and Case.III)***. As shown in Table 6, compared with Case.I, the proposed DHA module in EFF-Net not only reduces FLOPs and parameters by **60.32%** and **51.74%**, but also improves PSNR by **0.74 dB** and **0.79 dB** on Urban100 and SIDD validation datasets, respectively. This shows that DHA module can reduce FLOPs and the amount of parameters and improve the generalization performance of image denoising. With similar FLOPs and number of parameters, compared with Case.II and Case.III, the proposed EFF-Net achieves **0.32 dB** and **0.21 dB** improvement of average PSNR on validation datasets, respectively. This demonstrates that the proposed DHA module effectively suppresses the negative impact of low-weight tokens on image denoising performance, thereby improving image denoising performance.

To further explore the structural settings of the DF module and the influence of the learnable threshold on the image denoising performance, we conduct detailed ablation experiments on these above factors, respectively. As shown in Table 7, we conduct ablation experiments on the branches that decompose high-/medium-/low-frequencies in the DF module, the order of the branches, and the tensor operations between the branches. In Table 7, we find that when the original structure in the DF module is replaced with three branches (the dilation rate d of each branch is the same), the image denoising performance drops significantly. In addition, we change the order of branches (*i.e.*, the dilation rate d in the branches), and the image denoising performance is also weakened at this time. Finally, we replace the tensor operation between branches from subtraction to addition, and the image denoising performance on both test sets also declines. This shows that the structure and settings of the DF module are effective for image denoising performance.

To more intuitively illustrate the suppression of low-weight tokens by the DHA module, we visualize the feature maps of Case.I, Case.II, Case.III and the DHA module respectively, as shown in Fig. 9. Compared with the feature maps of Case.I, Case.II and Case.III, the area where the low-weight value of the tokens is dynamically zeroed by the Hash layer has obvious noise characteristics, demonstrating that weakening the low-weight tokens can alleviate the negative effects on reconstructed denoised images.

It can be seen that the DHA module essentially achieves the zeroing or suppression of regions with obvious noise characteristics through the Hash layer. This indicates that the proposed DHA module can effectively alleviate the negative impact of low-weight tokens on image denoising performance to improve the visual quality of reconstructed denoised images.

4.4.3. Ablation study of the EFF module

Since the core component of the EFF module is the Disassemble Frequency (DF) module, it is essentially the ablation study of the DF module. To study the impact of DF module on image denoising, we set up the DF module in EFF module and use Discrete Fourier Transform (DFT) [48], Discrete Cosine Transform (DCT) [49] and Discrete Wavelet Transform (DWT) [50] to replace the DF module scheme (denoted as Case.IV, Case.V and Case.VI). As shown in Table 8, compared with Case.IV, Case.V and Case.VI, the proposed DF module not only reduces FLOPs by **81.19%**, **64.69%** and **53.26%**, respectively,

Table 7
Effects of branching, branching order, and tensor operations in the DF module on image denoising performance.

Case	Settings	Urban100 (PSNR:dB)	SIDD (PSNR:dB)
Three branches (kernel=3)	$d = 1$	34.68	39.76
	$d = 2$	34.52	39.64
	$d = 3$	34.39	39.50
Branch order	$d = 3, d = 2, d = 1$	35.01	39.97
	$d = 3, d = 1, d = 2$	34.93	39.82
	$d = 2, d = 1, d = 3$	34.88	39.75
	$d = 2, d = 3, d = 1$	34.99	39.94
	$d = 1, d = 3, d = 2$	34.82	39.71
	$d = 1, d = 2, d = 3$ (Ours)	**35.27**	**40.15**
Tensor operations in DF module	addition	34.98	39.81
	subtraction (Ours)	**35.27**	**40.15**

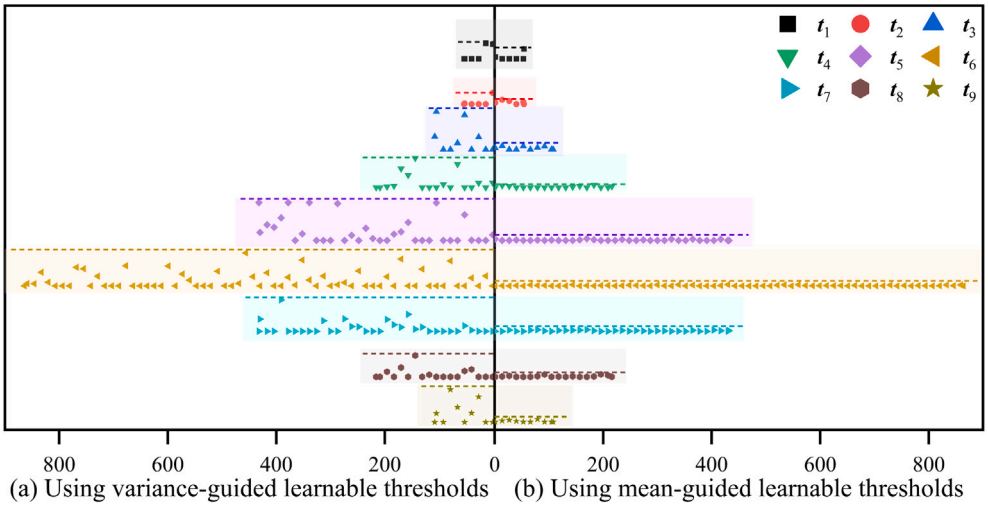

Fig. 10. Visual comparisons between EFF-Net and its competitors in the evaluation of real-noisy image denoising. Abscissa represents the ith of feature map channels in each AFE block. Ordinate represents the value range of the threshold t_i.

Table 8
Performance effect of the DHA module on image denoising. The PSNR (dB) is used as a reference metric for image denoising performance on the validation dataset.

Method	FLOPs	#Prams.	Urban100 (PSNR:dB)	SIDD (PSNR:dB)
Case.IV: DFT	108.32 G	10.28 M	33.97	38.41
Case.V: DCT	57.69 G	10.28 M	34.39	39.06
Case.VI: DWT	43.58 G	10.28 M	34.83	39.89
EFF-Net (Ours)	**20.37 G**	10.28 M	**35.27**	**40.15**

Table 9
Effects of threshold types and settings on image denoising performance.

Case	Settings	Urban100 (PSNR:dB)	SIDD (PSNR:dB)
Fixed threshold	$t = 0.1$	35.12	39.86
	$t = 0.2$	35.01	39.75
	$t = 0.3$	34.89	39.63
	$t = 0.4$	34.77	39.51
	$t = 0.5$	33.83	38.28
Learnable threshold	variance	34.05	38.56
	mean (Ours)	**35.27**	**40.15**

but also improves PSNR on average by **1.52 dB**, **0.97 dB** and **0.35 dB** on the validation dataset. This demonstrates that the proposed DF module can effectively separate different frequency band features for subsequent adaptive enhancement (*i.e.*, using only one convolutional layer) to improve image denoising performance.

To further explore the influence of the learnable threshold on the image denoising performance, we conduct detailed ablation experiments on this above factor. As shown in Table 9, we conduct ablation experiments on fixed thresholds and thresholds learned by using the values of statistical properties of feature maps (*i.e.*, variance and mean). Since each AFE block contains a learnable threshold t, EFF-Net contains nine different learnable thresholds $\boldsymbol{t}$ (*i.e.*, t_i, $i = 1, 2, \ldots, 9$). To observe the effect of fixed threshold on image denoising performance, we simplified the fixed threshold in EFF-Net, that is, each threshold $\boldsymbol{t}$ is the same. It can be found in Table 9 that the image denoising performance using a fixed threshold is significantly lower compared to the learnable threshold. This is because the fixed threshold cannot be changed with the change of the sample data, hence, it has great limitations. Furthermore, we perform an ablation study on learnable thresholds with variance-guided learning, which is a common statistical property of feature maps. Furthermore, we perform an ablation study on learnable thresholds with variance-guided learning, which is a common statistical property of feature maps. In Table 9, we find that image denoising performance is significantly higher using mean-guided learnable thresholding compared to using variance-guided learnable thresholding. As shown in Fig. 10, using variance-guided learnable thresholds is more discrete than using mean-guided learnable thresholds, which is greatly easy to over-zero tokens with low weight values, resulting in poor image denoising performance.

5. Conclusion

In this paper, we propose a **F**requency **A**daptive **E**nhancement **Net**work (EFF-Net) with dynamic hash attention for image denoising tasks. Specifically, to mitigate that tokens with low-weight values may negatively affect the reconstructed denoised images, we propose a Dynamic Hash Attention (DHA) module, which can effectively mitigate tokens with low weight values negative impact on image denoising performance. Moreover, in order to enhance the texture and details of the

reconstructed image, we designed an **E**nhanced **F**requency **F**usion (EFF) module with the **D**ecomposition **F**requency (DF) as the core component, which can achieve adaptive enhancement of different frequency components to improve the visual quality of the reconstructed denoised image. Then, the DHA and FE modules are further integrated into a plug-and-play **A**daptive **F**requency **E**nhancement (AFE) transformer block to achieve the dynamic reconstruction of image denoising by incorporating long-range pixel dependencies. Through extensive and comprehensive experiments on denoising tasks, the proposed EFF-Net achieves state-of-the-art results, demonstrating satisfactory superiority in denoising and efficiency.

CRediT authorship contribution statement

Bo Jiang: Conceptualization, Methodology, Software, Writing – original draft, Visualization, Formal analysis, Validation. **Jinxing Li:** Writing – review, Validation, Supervision, Funding acquisition, Project administration. **Huafeng Li:** Writing – review. **Ruxian Li:** Writing – review. **David Zhang:** Writing – review. **Guangming Lu:** Writing – review & editing, Supervision, Project administration, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported in part by National Key Research and Development Program of China under Project Number 2018AAA0100100, in part by the NSFC fund 62176077, in part by the Shenzhen Key Technical Project under Grant 2022N001, 2020N046, in part by the Shenzhen Fundamental Research Fund under Grant JCYJ20210324132210025, and in part by the Medical Biometrics Perception and Analysis Engineering Laboratory, Shenzhen, China. Also, this work was supported in part by the NSFC fund (62272133), the Shenzhen Colleges and Universities Stable Support Program (GXWD20220811170100001) and Shenzhen Science and Technology Program (RCBS20200714114910193).

References

[1] B. Goyal, A. Dogra, S. Agrawal, B.S. Sohi, A. Sharma, Image denoising review: From classical to state-of-the-art approaches, Inf. Fusion 55 (2020) 220–244.
[2] Y. Wu, S. Li, A novel fusion paradigm for multi-channel image denoising, Inf. Fusion 77 (2022) 62–69.
[3] S. Xu, J. Zhang, J. Wang, K. Sun, C. Zhang, J. Liu, J. Hu, A model-driven network for guided image denoising, Inf. Fusion 85 (2022) 60–71.
[4] Y. Gan, T. Xiang, H. Liu, M. Ye, Learning-aware feature denoising discriminator, Inf. Fusion (2022).
[5] S. Miller, C. Zhang, K. Hirakawa, Multi-resolution aitchison geometry image denoising for low-light photography, IEEE Trans. Image Process. 30 (2021) 5724–5738.
[6] K. Zhang, W. Zuo, Y. Chen, D. Meng, L. Zhang, Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising, IEEE Trans. Image Process. 26 (2017) 3142–3155.
[7] K. Zhang, W. Zuo, L. Zhang, FFDNet: Toward a fast and flexible solution for CNN-based image denoising, IEEE Trans. Image Process. 27 (2018) 4608–4622.
[8] B. Kim, J. Lee, J. Kang, E. Kim, H.J. Kim, HOTR: end-to-end human-object interaction detection with transformers, in: IEEE Conference on Computer Vision and Pattern Recognition, 2021, pp. 74–83.
[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16×16 words: Transformers for image recognition at scale, in: International Conference on Learning Representations, 2021.
[10] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, H. Jégou, Training data-efficient image transformers & distillation through attention, in: International Conference on Machine Learning, 2021, pp. 10347–10357.
[11] W. Wang, E. Xie, X. Li, D. Fan, K. Song, D. Liang, T. Lu, P. Luo, L. Shao, Pyramid vision transformer: A versatile backbone for dense prediction without convolutions, 2021, CoRR abs/2102.12122.
[12] Z. Wang, X. Cun, J. Bao, J. Liu, Uformer: A general u-shaped transformer for image restoration, 2021, arXiv preprint arXiv:2106.03106.
[13] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, R. Timofte, Swinir: Image restoration using swin transformer, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 1833–1844.
[14] S.W. Zamir, A. Arora, S. Khan, M. Hayat, F.S. Khan, M.-H. Yang, Restormer: Efficient transformer for high-resolution image restoration, 2021, arXiv preprint arXiv:2111.09881.
[15] S.G. Mallat, A theory for multiresolution signal decomposition: The wavelet representation, IEEE Trans. Pattern Anal. Mach. Intell. 11 (7) (1989) 674–693.
[16] M. Aharon, M. Elad, A.M. Bruckstein, $rm K$-SVD: An algorithm for designing overcomplete dictionaries for sparse representation, IEEE Trans. Signal Process. 54 (2006) 4311–4322.
[17] A. Buades, B. Coll, J.-M. Morel, A non-local algorithm for image denoising, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, CVPR'05, 2005, pp. 60–65.
[18] K. Dabov, A. Foi, V. Katkovnik, K.O. Egiazarian, Image denoising by sparse 3-D transform-domain collaborative filtering, IEEE Trans. Image Process. 16 (2007) 2080–2095.
[19] D. Zoran, Y. Weiss, From learning models of natural image patches to whole image restoration, in: 2011 International Conference on Computer Vision, 2011, pp. 479–486.
[20] B. Jiang, Y. Lu, J. Wang, G. Lu, D. Zhang, Deep image denoising with adaptive priors, IEEE Trans. Circuits Syst. Video Technol. (2022).
[21] B. Jiang, Y. Lu, G. Lu, D. Zhang, Real noise image adjustment networks for saliency-aware stylistic color retouch, Knowl.-Based Syst. 242 (2022) 108317.
[22] Z. Zha, X. Yuan, J. Zhou, C. Zhu, B. Wen, Image restoration via simultaneous nonlocal self-similarity priors, IEEE Trans. Image Process. 29 (2020) 8561–8576.
[23] Z. Zha, X. Yuan, B. Wen, J. Zhou, J. Zhang, C. Zhu, From rank estimation to rank approximation: Rank residual constraint for image restoration, IEEE Trans. Image Process. 29 (2019) 3254–3269.
[24] S. Guo, Z. Yan, K. Zhang, W. Zuo, L. Zhang, Toward convolutional blind denoising of real photographs, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 1712–1722.
[25] S. Anwar, N. Barnes, Real image denoising with feature attention, in: 2019 IEEE/CVF International Conference on Computer Vision, ICCV, 2019, pp. 3155–3164.
[26] S.W. Zamir, A. Arora, S.H. Khan, M. Hayat, F.S. Khan, M.-H. Yang, L. Shao, Learning enriched features for real image restoration and enhancement, 2020, ArXiv abs/2003.06792.
[27] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y.R. Fu, Residual dense network for image super-resolution, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 2472–2481.
[28] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).
[29] N. Nangia, A. Williams, A. Lazaridou, S.R. Bowman, The RepEval 2017 shared task: Multi-genre natural language inference with sentence representations, in: RepEval@EMNLP, 2017.
[30] Z. Chen, H. Zhang, X. Zhang, L. Zhao, Quora question pairs, 2017.
[31] E. Hulburd, Exploring BERT parameter efficiency on the stanford question answering dataset v2.0, 2020, ArXiv abs/2002.10670.
[32] A. Mahmoud, M. Zrigui, Arabic semantic textual similarity identification based on convolutional gated recurrent units, in: 2021 International Conference on INnovations in Intelligent SysTems and Applications, INISTA, 2021, pp. 1–7.
[33] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, N. Houlsby, An image is worth 16×16 words: Transformers for image recognition at scale, 2021, ArXiv abs/2010.11929.
[34] H. Touvron, M. Cord, H. J'egou, DeiT III: Revenge of the ViT, 2022, ArXiv abs/2204.07118.
[35] F. Wang, H. Wang, C. Wei, A.L. Yuille, W. Shen, CP2: Copy-paste contrastive pretraining for semantic segmentation, 2022, ArXiv abs/2203.11709.
[36] J. Chen, Y. Lu, Q. Yu, X. Luo, E. Adeli, Y. Wang, L. Lu, A.L. Yuille, Y. Zhou, TransUNet: Transformers make strong encoders for medical image segmentation, 2021, ArXiv abs/2102.04306.
[37] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, S. Zagoruyko, End-to-end object detection with transformers, 2020, ArXiv abs/2005.12872.
[38] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, J. Dai, Deformable DETR: Deformable transformers for end-to-end object detection, 2021, ArXiv abs/2010.04159.
[39] H. Chen, Y. Wang, T. Guo, C. Xu, Y. Deng, Z. Liu, S. Ma, C. Xu, C. Xu, W. Gao, Pre-trained image processing transformer, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2021, pp. 12294–12305.
[40] K. Zhang, Y. Li, J. Liang, J. Cao, Y. Zhang, H. Tang, R. Timofte, L.V. Gool, Practical blind denoising via Swin-Conv-UNet and data synthesis, 2022, ArXiv abs/2203.13278.

[41] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, B. Guo, Swin transformer: Hierarchical vision transformer using shifted windows, in: 2021 IEEE/CVF International Conference on Computer Vision, ICCV, 2021, pp. 9992–10002.
[42] S.M. Bellovin, E.K. Rescorla, Deploying a new hash algorithm, 2005.
[43] T. Zhan, S. Chen, An improved hash algorithm for monitoring network traffic in the Internet of Things, Cluster Comput. (2022) 1–16.
[44] Y. Li, J. Ma, Y. Zhang, Image retrieval from remote sensing big data: A survey, Inf. Fusion 67 (2021) 94–115.
[45] X. Luo, Y. Xie, Y. Zhang, Y. Qu, C. Li, Y. Fu, Latticenet: Towards lightweight image super-resolution with lattice block, in: European Conference on Computer Vision, Springer, 2020, pp. 272–289.
[46] Y. Chen, H. Fan, B. Xu, Z. Yan, Y. Kalantidis, M. Rohrbach, S. Yan, J. Feng, Drop an octave: Reducing spatial redundancy in convolutional neural networks with octave convolution, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3435–3444.
[47] F. Yu, V. Koltun, Multi-scale context aggregation by dilated convolutions, 2016, CoRR abs/1511.07122.
[48] S. Winograd, On computing the discrete Fourier transform, Math. Comp. 32 (141) (1978) 175–199.
[49] N. Ahmed, T. Natarajan, K.R. Rao, Discrete cosine transform, IEEE Trans. Comput. 100 (1) (1974) 90–93.
[50] T. Edwards, Discrete wavelet transforms: Theory and implementation, Universidad de (1991) 28–35.
[51] P. Charbonnier, L. Blanc-Féraud, G. Aubert, M. Barlaud, Two deterministic half-quadratic regularization algorithms for computed imaging, in: Proceedings of 1st International Conference on Image Processing, vol. 2, 1994, pp. 168–172.
[52] I. Loshchilov, F. Hutter, Fixing weight decay regularization in Adam, 2017, ArXiv abs/1711.05101.
[53] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in: 2015 IEEE International Conference on Computer Vision, ICCV, 2015, pp. 1026–1034.
[54] S. Roth, M.J. Black, Fields of experts: A framework for learning image priors, in: 2005 IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, CVPR, 2005, pp. 860–867.
[55] J.-B. Huang, A. Singh, N. Ahuja, Single image super-resolution from transformed self-exemplars, in: 2015 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2015, pp. 5197–5206.
[56] K. Zhang, W. Zuo, S. Gu, L. Zhang, Learning deep CNN denoiser prior for image restoration, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2017, pp. 2808–2817.
[57] K. Zhang, W. Zuo, L. Zhang, FFDNet: Toward a fast and flexible solution for CNN-based image denoising, IEEE Trans. Image Process. 27 (2018) 4608–4622.
[58] P. Liu, H. Zhang, K. Zhang, L. Lin, W. Zuo, Multi-level wavelet-CNN for image restoration, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, CVPRW, 2018, pp. 886–88609.
[59] D. Liu, B. Wen, Y. Fan, C.C. Loy, T.S. Huang, Non-local recurrent network for image restoration, 2018, ArXiv abs/1806.02919.
[60] Y. Zhang, K. Li, K. Li, B. Zhong, Y.R. Fu, Residual non-local attention networks for image restoration, 2019, ArXiv abs/1903.10082.
[61] X. Jia, S. Liu, X. Feng, L. Zhang, FOCNet: A fractional optimal control network for image denoising, in: 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2019, pp. 6047–6056.
[62] C. Mou, J. Zhang, Z. Wu, Dynamic attentive graph learning for image restoration, in: 2021 IEEE/CVF International Conference on Computer Vision, ICCV, 2021, pp. 4308–4317.
[63] K. Zhang, Y. Li, W. Zuo, L. Zhang, L.V. Gool, R. Timofte, Plug-and-play image restoration with deep denoiser prior, IEEE Trans. Pattern Anal. Mach. Intell. PP (2021).
[64] W. Li, X. Lu, J. Lu, X. Zhang, J. Jia, On efficient transformer and image pre-training for low-level vision, 2021, ArXiv abs/2112.10175.
[65] R. Franzen, Kodak lossless true color image suite, 1999, Source: http://r0k.us/graphics/ko 4 (2).
[66] S.M. Kasar, S. Ruikar, Image demosaicking by nonlocal adaptive thresholding, in: 2013 International Conference on Signal Processing, Image Processing & Pattern Recognition, 2013, pp. 34–38.
[67] C. Tian, Y. Xu, W. Zuo, Image denoising using deep CNN with batch renormalization, Neural Netw. 121 (2020) 461–473.
[68] A. Abdelhamed, S. Lin, M.S. Brown, A high-quality denoising dataset for smartphone cameras, in: 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018, pp. 1692–1700.
[69] J. Xu, H. Li, Z. Liang, D.C. Zhang, L. Zhang, Real-world noisy image denoising: A new benchmark, 2018, ArXiv abs/1804.02603.
[70] S. Nam, Y. Hwang, Y. Matsushita, S.J. Kim, A holistic approach to cross-channel image noise modeling and its application to image denoising, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR, 2016, pp. 1683–1691.
[71] J. Xu, L. Zhang, D.D. Zhang, A trilateral weighted sparse coding scheme for real-world image denoising, 2018, ArXiv abs/1807.04364.
[72] Z. Yue, H. Yong, Q. Zhao, L. Zhang, D. Meng, Variational denoising network: Toward blind noise modeling and removal, in: NeurIPS, 2019.
[73] R. Ma, B. Zhang, Y. Zhou, Z. Li, F. Lei, PID controller-guided attention neural network learning for fast and effective real photographs denoising, IEEE Trans. Neural Netw. Learn. Syst. PP (2021).
[74] Y. Kim, J.W. Soh, G.Y. Park, N.I. Cho, Transfer learning from synthetic to real-noise denoising with adaptive instance normalization, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR, 2020, pp. 3479–3489.
[75] R.-D. Ma, S. Li, B. Zhang, Z. Li, Towards fast and robust real image denoising with attentive neural network and PID controller, IEEE Trans. Multimed. (2021) 1.
[76] J. Hu, L. Shen, S. Albanie, G. Sun, E. Wu, Squeeze-and-excitation networks, IEEE Trans. Pattern Anal. Mach. Intell. 42 (2020) 2011–2023.
[77] M. Jaderberg, K. Simonyan, A. Zisserman, K. Kavukcuoglu, Spatial transformer networks, in: NIPS, 2015.